NET ART && CULTURES

FVNMA 3235-001 (1485)

where:	Michigan 807 ;
when:	Tu 9am - 4pm ;
professor:	Nick Briz ;
email:	nbriz@saic.edu ;
office hours:	*by-appointment-only* ;
class website:	https://github.com/net-art-and-cultures ;

// Course Description

This studio course examines the Internet as an artistic medium (computers, networks and code) as well as a masterpiece of human civilization (a la Virginia Heffernan); various forms of online culture will be shared, discussed and produced. While we will occasionally be discussing canonical Internet Art (jodi.org, Olia Lialina, Alexi Shulgin, etc.) we will generally be taking a broader cultural view, also considering the works of web designers/developers, meme makers, demoscensters and hackers. This course is also an introduction to programming (beginner level JavaScript), a large emphasis will be placed on developing technical proficiencies related to the web (intermediate level HTML/CSS, Browser APIs and developer tools). The technologies studied will also be discussed from an ethical perspective (Internet health, machine bias, power, access, control) understanding how they work as environments (media ecology) and how they inform on/offline communities.

// Prerequisite

The prerequisite for this class is New Media Crash Course (FVNM 2100). This is an intermediate to advanced level studio course, students should arrive with a general understanding of what the Internet is and how it works (specifically the web). They should have beginner level HTML/CSS knowledge.

// Materials and Supplies

Students are expected to bring their laptops to class every week. They may also be required to purchase a Raspberry Pi computer (~$35) and the accompanying power supply and SD card (but this is pending which experiments we end up working on in class, see “experiments” section at the end of the syllabus). It is also recommended that you purchase a grid-paper notebook (though it is not required, see “notes” section at the end of the syllabus)

Students are also expected to have downloaded and installed the following applications by the second week of class: Atom, FileZilla, git, Firefox and nodejs. links/instructions for downloading all these applications is detailed on the class website (https://github.com/net-art-and-cultures/syllabus-and-notes/blob/master/supplies.md)

Students will also be expected to create accounts on github, Digital Ocean, Namecheap and StackOverflow (~$50 total, or $0 with credit available through github student dev pack)

All readings, watchings && listenings will be provided by the instructor. Notes and resources will be available on the class website: (https://github.com/net-art-and-cultures/syllabus-and-notes/).

// Accommodations for Students with Disabilities

Accommodations for Students with Disabilities:SAIC is committed to full compliance with all laws regarding equal opportunities for students with disabilities.Students with known or suspected disabilities, such as a Reading/Writing Disorder, ADD/ADHD, and/or a mental health condition who think they would benefit from assistance or accommodations should first contact the Disability and Learning Resource Center (DLRC) to schedule an appointment. DLRC staff will review your disability documentation and work with you to determine reasonable accommodations. They will then provide you with a letter outlining the approved accommodations for you to deliver to your instructors.This letter must be presented before any accommodations will be implemented. You should contact the DLRC as early in the semester as possible.The DLRC is located within the Wellness Center on the 13th floor of 116 S Michigan Ave. and can be reached via phone at 312.499.4278 or email atdlrc@saic.edu.

// Attendance

SAIC policy states that students are expected to attend all classes regularly and on time. if a student arrives to class more than 15 minutes after 9am or leaves more than 15 minutes before 4pm it will be considered a half absence.

Students should miss class only with reasonable cause. **If a student needs to miss class with reasonable cause, it is the student's responsibility to contact the professor before the date of the class being missed to receive instruction for how to make up for the missed class.** If the student misses a class for other than a reasonable cause (communicated prior to the absence) the student will fail the class. **If a student misses MORE than three (excused) classes, the student will fail the class. Add/Drop deadline: Tue, Sept 10th; deadline for withdrawal: Tue, October 29, 2019.**

Reasonable cause to miss a class might include:

- ➔ Illness or hospitalization (the student should contact Health Services, who will relay information to the faculty in whose class the student is enrolled)
- ➔ Family illness or death
- ➔ professional opportunities (should be communicated to the professor && approved ahead of time)

// Learning Goals

- ★ Solid understanding of what the Internet is and how it works (specifically the web)
- ★ General background on key cultural/historical figures/movements and motivations which have informed the development of online culture.
- ★ Foundational understanding of web programming concepts, tools && working knowledge the JavaScript programming language.
- ★ Solid understanding of what it means to work collaboratively on online projects including a working knowledge of open-source collaboration tools
- ★ General understanding of the political/social/aesthetic implications of living in a networked culture (*what copying/collaboration/authorship mean today; what algorithms are and how they affect us; what privacy means in a surveillance state, a general understanding of the economic/sociopolitical role networks && data play in our lives*)

// Evaluation

credit will be determined according to the following breakdown:

- ➔ participation in discussions ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ 20%
- ➔ notes ~ 15%
- ➔ experiments ~ 20%
- ➔ midterm pitches ~ 20%
- ➔ final collaborative project ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ 25%

specific details on how notes/sketches, experiments and the final project are evaluated are at the end of the syllabus

Courses for which a student registers are recorded on the student's permanent record. SAIC adheres to a credit/no credit grading system.

The adjacent grade symbols are used to denote credit status. Undergraduate and non-degree-seeking students must achieve at least average performance in the course the traditional grade equivalent of a C or 80% in order to earn CR (credit). Graduate students must achieve the traditional grade equivalent of a B or 90% in order to earn CR (credit).

If a student would like a grade equivalent they should give you a Student Letter Grade Form, which they can request at the Registrar's Office.

CR	Credit
NCR	No Credit
W	Withdrawal
INC	Incomplete
IP	Thesis In Progress
NR	Grade Not Reported by Instructor

Grades of INC (Incomplete) will be granted by the instructor only upon request by the student and only if the instructor believes that the student's reason for the request is justified. Incompletes must be completed within the first two weeks of the next regular semester, or the grade will automatically be changed to NCR (no credit).

// Class Discussions

So much of what we're going to cover in class, both in terms of the theory and practice, can be gleaned through your own online research. Perhaps the most valuable aspect of learning this material in the classroom, rather than on your own, is the chance for realtime interactivity with your professor and peers. I can not stress enough how important it is to take advantage of class discussions. In order to fully participate in these discussions you must be present, on time and attentive. You must also complete any/all readings (or videos/podcasts) assigned for homework as well as maintain good notes (see below).

// Notes

You will be expected to keep organized notes throughout the semester. These can be written down in a physical notebook (ex: grid-paper journal mentioned in the class materials section above) or in a digital document on your computer (as long as you bring it to class). In addition to the obvious/general notes taken in class during lectures or at home during your own research/experimentation time, there are two important *kinds* of notes I will be expecting you to bring to class on a weekly basis.

The first is **questions:** any thoughts/curiosities, whether technical or theoretical, based on a homework assignment or generally related to the subjects covered in class should be written down in your notes. Maybe there were references made in a reading assignment you didn't understand and had a hard time looking up yourself? Maybe someone shared an article on social media about a data leak or online surveillance you don't fully understand? Maybe there's a bug you keep getting and can't seem to find the solution for online? Write them all down and bring these to class. That said, if at any point you are having an issue (say with a homework assignment) that's private/personal or otherwise can't wait for a class you are of course encouraged to email me (nbriz@saic.edu) at any point throughout the semester.

The second is **ideas:** specifically, ideas for online projects (however vague or specific, unconventional or derivative), this will ensure you are regularly thinking about the creative potential of the Internet as well as become a starting points for your midterm proposal (see below). They may also become the basis for class experiments/workshops/discussions. Don't worry about whether or not an idea you have if feasible, we can always discuss the viability of any idea in class or via email (but a general rule of thumb: if you can imagine it online then it's most probably possible).

// Experiments

You will be completing a handful of experiments during the first half of the semester (before shifting focus onto the final project during the second half). The specifics of these experiments will depend on the technical literacy of the students enrolled in the class and the speed at which we cover the material. But for context, these may include: a personal homepage hosted on Github pages, CSS still lifes, "Form Art" portraits, "Pure CSS" drawings, algorithmic musical compositions, virtual online environment (3D/VR), ASCII/ANSI Art Bulletin

Board System (BBS), Guerilla AP (DIY network running on a Raspberry Pi). Each experiment will be submitted via Github (with a couple of exceptions) and will be evaluated based on specific criteria which will accompany each assignment.

// Midterm Pitches

Everyone will conceive of and present ideas for web based “artware” (software art) projects. We’ll draw inspiration from Matthew Fuller’s categories of non-utilitarian software (social software, critical software, speculative software) as well as the various artware projects/examples discussed in class. Together as a class we will choose one (or combine a couple) of the ideas as the starting point for our collective final project.

// Final Project

The final project will be a collaborative “open-source” style Internet Art project. The details of which are to be determined (as we will be designing it together in class). Your contribution to the project will be submitted as a github “pull-request” and evaluations will be made on the actual “pull-request” (the motivation for and details on how this works will be learned and explained in class). the technical criteria your pull-request will be evaluated on is as follows:

- ➔ Did you work **collaboratively**? Is your contribution informed by class discussions and any feedback you received on github from your collaborators?
- ➔ Is the source **code** properly **open-source**? Meaning not only is available on github, but is it clean? Is it well organized and documented? Is it easy to read and contribute to? are you following the style/conventions we collaboratively agreed on in class?
- ➔ Is your code properly **attributed**? Are libraries and other code bases you use properly credited in your documentation? Does your source code include links to online posts where you found solution to tiny problems/functions/etc?
- ➔ Did you **version** your project well? Did you make consistent git “commits” is it easy to traverse the history of your creative/development process?
- ➔ Is your code written **modularly**? If there are portions of your code which could be remixed, used on other projects in different ways, was this code written in a way as to intentionally facilitate that appropriation?